

# 기초웹개발론

하진수

<https://goo.gl/R47xXb>

Kakao

지난번 연습 문제

<https://goo.gl/WeCRWv>

# CSS part.1

# 스타일을 사용하는 이유?

- 웹 문서의 내용과 상관없이 디자인만 바꿀 수 있음
  - <http://www.csszengarden.com/>
- 다양한 기기에 맞게 탄력적으로 바뀌는 문서를 만들 수 있음

# 스타일 형식

```
/* 텍스트 단락을 중앙에 정렬하는 스타일 규칙 */  
p { text-align : center; }
```

선택자

스타일 속성

속성 값

# 스타일 표기 방법

- 스타일 속성은 세미콜론(;)으로 구분해 중괄호 안에 나열
- 파일 크기를 작게하기 위한 목적이 아니면 각 속성은 여러줄에 나누어 표기(공백은 무시됨)
- 주석은 /\*와 \*/ 사이에 입력

# 스타일과 스타일 시트

- 스타일 시트 : 여러개의 스타일 규칙들을 한 군데 묶어 놓은 것
- 내부 스타일 시트
  - 웹 문서 안에 스타일을 정리
  - `<style>`태그와 `</style>` 태그 사이에 내용을 작성하고, `<head>` `</head>` 태그 안에 `<style>``</style>` 태그를 위치
- 외부 스타일 시트
  - `<link href="외부 스타일 시트 파일 경로" rel="stylesheet" type="text/css">`

# 인라인 스타일

- 간단한 스타일 정보인 경우 태그에 스타일을 직접 표시
- <tag style="속성:속성값;"> 형태
- <span style="color:blue;font-weight:bold">



# 스타일 선택자

선택자	형식
전체 선택자	* {속성: 속성값; 속성: 속성값; ...}
태그 선택자	태그 {속성: 속성값; ...}
클래스 선택자	.클래스명 {속성: 속성값; ...}
id 선택자	#아이디 {속성: 속성값; ...}
그룹 선택자	이름1, 이름2 {속성: 속성값; ...}

# 복습

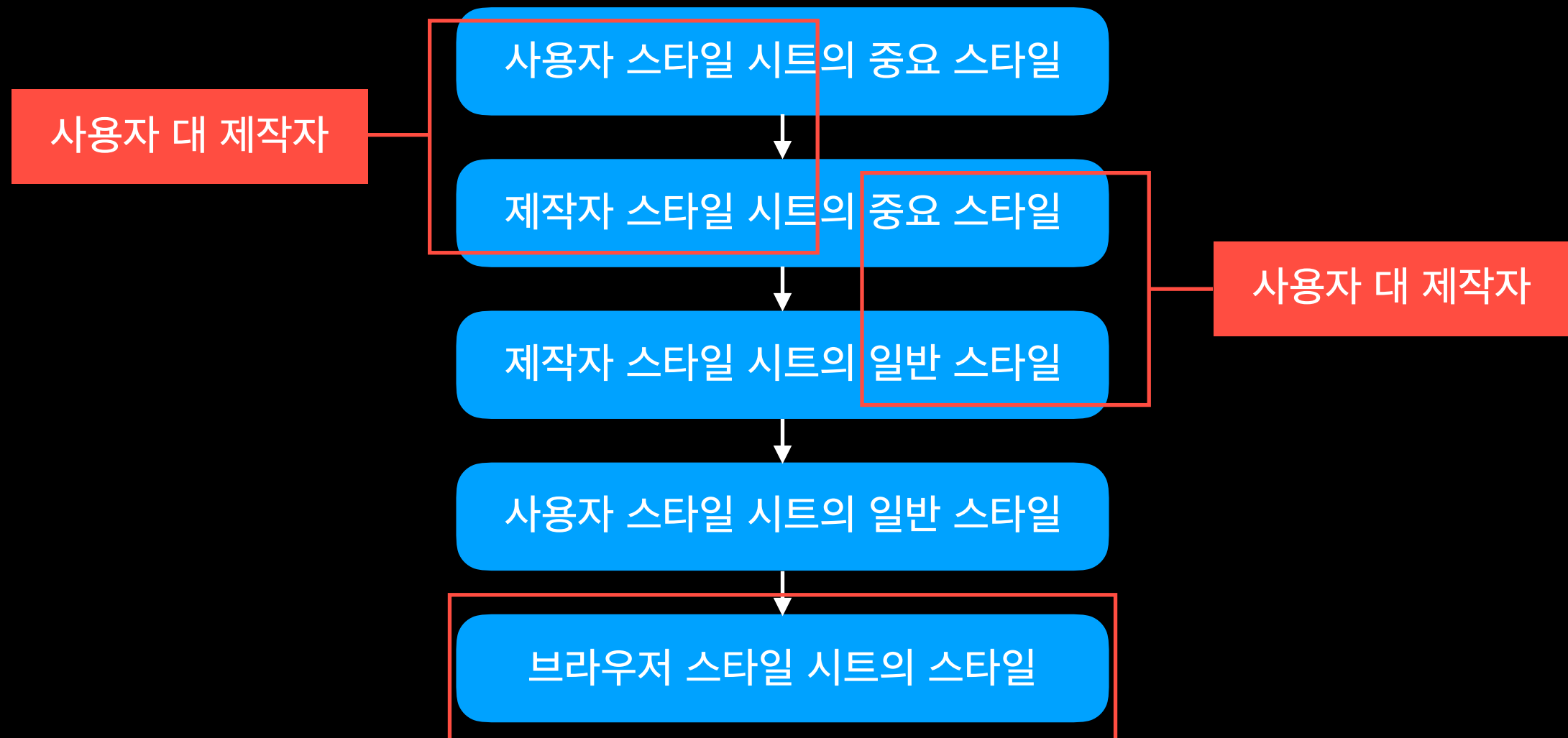
```
<style>
  [redacted] { text-decoration: underline; }
</style>
<p>문서의 특정 클래스에 스타일을 적용할 때는
<span [redacted]="underline">클래스 선택자</span>와
<span [redacted]="underline">id 선택자</span>를 사용합니다.</p>
```

# Cascading Style Sheets

- 위에서 아래로 흐르는 스타일 시트
- 선택자에 적용된 많은 스타일 중 어떤 스타일을 나타낼지 결정함을 뜻함
- 스타일 간의 충돌을 막기 위한 방법이 cascading
- 2가지 원칙
  - 스타일 우선 순위
  - 스타일 상속

# 스타일 우선 순위

- 어떤 스타일을 먼저 적용할지 결정하는 규칙



Kakao

# 사용자 vs 제작자

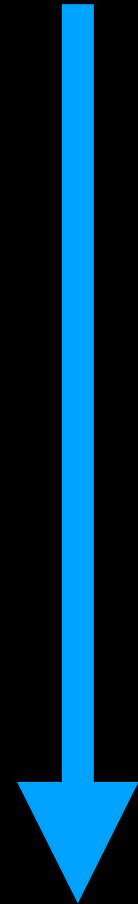
- 사용자 스타일 시트가 우선
  - 저시력자나 색약자 등과 같이 특별한 조건이 필요한 사용자가 사용자에 맞게 구성해 놓은 스타일 시트
- ex. 고대비 설정
- 이 스타일 시트는 시스템을 통해 만들어진 것이기 때문에 제작자가 제어할 수 없음

# 중요 스타일

- 일반 스타일
  - 스타일 시트에서 작성하는 스타일 규칙
- !important
  - 다른 어떤 스타일보다 최우선으로 적용해야할 스타일
  - 스타일 뒤에 붙임

# 명시도 - Specificity

- 인라인 스타일
  - 태그 안에 style 속성으로 해당 태그에만 적용
- id 스타일
  - 지정한 부분에만 적용되는 스타일. 한 문서 안에서 한 번만 적용
- 클래스 스타일
  - class 속성과 일치되는 부분에만 적용
- 태그 스타일
  - 특정 태그에 똑같이 적용되는 스타일



# 스타일 상속

- 스타일 시트에서는 자식 요소에 별도로 스타일을 지정하지 않는 경우 부모의 스타일 속성들이 자식 요소로 전달됨
- 상속을 이용해서 스타일 시트를 효과적으로 만들 수 있음
  - ex. 모든 요소에 같은 폰트를 사용하려고 할 때 각 요소마다 지정하지 않고, body 태그에 스타일을 작성
- 모든 속성이 상속되지는 않음 (배경색, 배경 이미지 등)



# CSS3

- CSS3 Module
  - 배경, 글꼴, 박스 모델 등 수십 개의 기능을 주제별로 따로 만들었음
  - 모듈 중 일부는 완성된 것도 있고, 아직 개발중인 것도 있음
  - 브라우저에 따라 지원되는 기능도 있고 지원되지 않은 기능도 있음
- CSS3는 'CSS3 표준'이라고 하나로 묶어 말할 수 없고 계속 새로운 CSS 모듈이 등장할 수 있음

# CSS3 prefix

- 표준 규약이 아닌 속성들은 브라우저에 따라 다른 방식으로 지원
- 속성앞에 접두사(prefix)를 붙여 브라우저별로 구분
- 표준 규약이 만들어졌더라도 이전 버전의 모던 브라우저 사용자를 고려해 계속 브라우저 접두사를 붙여 사용하기도 함

접두사	설명
-webkit-	웹킷 방식 브라우저용(safari, 크롬 등)
-moz-	게코 방식 브라우저용(모질라, 파이어폭스 등)
-o-	오페라 브라우저
-ms-	MS 인터넷 익스플로러

# 텍스트 관련 스타일

# 글꼴 관련 스타일

- font-family — 글꼴 지정
  - font-family:<글꼴 이름>[,<글꼴 이름>,<글꼴 이름>]
  - 지정된 폰트가 시스템에 설치되어 있지 않으면 기본 폰트로 보여짐
- initial, inherit

# 웹 폰트

- 시스템에 폰트가 설치되어 있지 않은 사용자에게도 동일한 폰트를 보여주기 위한 방법
- 구글 웹 폰트
  - <https://fonts.google.com/earlyaccess>
  - <https://fonts.google.com/>
- 직접 폰트 업로드 후 사용 - 저작권 주의

# 웹 폰트

- 직접 업로드 시 사용 방법

```
@font-face {  
    font-family: 글꼴 이름;  
    src: url(글꼴 파일 경로) format(파일 유형);  
}
```

```
@font-face {  
    font-family: 'trana';  
    src: local('trana'),  
        url('trana.woff') format('woff'),  
        Url('trana.ttf') format('truetype');  
}  
  
.w-font {  
    font-family: 'trana', sans-serif;  
}
```

# 글자 크기

- font-size: <절대 크기> | <상대 크기> | <크기> | <백분율>

속성 값	설명
<절대 크기>	브라우저에서 지정한 글자 크기. xx-small, x-small, small, medium, large, x-large, xx-large
<상대 크기>	부모 요소의 글자 크기를 기준으로 표시. larger, smaller
<크기>	글자 크기 직접 지정
<백분율>	부모 요소의 글자 크기를 기준으로 해당하는 %를 계산해 표시

# 글자 크기

단위	설명
em	해당 글꼴의 대문자 M의 너비를 기준으로 크기를 조절
ex	x-height. 해당 글꼴의 소문자 x의 높이를 기준으로 크기를 조절
px	픽셀. 모니터에 따라 상대적 크기가 됨
pt	포인트. 일반 문서에서 많이 사용되는 단위



# 글자 굵기 지정

- font-weight: normal | bold | bolder | lighter | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900

속성 값	설명
normal	일반적인 형태. 기본값
bold   lighter   bolder	굵게, 가늘게, 더 굵게
100~900	400은 normal, 700은 bold에 해당. 숫자값을 조절해 더 세밀하게 글꼴 두께를 조절할 수 있음

# 글자 스타일

- font-style: normal | italic | oblique

속성 값	설명
normal	일반적인 형태. 기본값
italic	이탤릭체
oblique	이탤릭체

# 연습

웹 문서에 포함된 텍스트들은 **사용자의 글꼴을 이용해**  
웹 브라우저 화면에 표시됩니다.

크기 : 20px  
스타일 : 이탤릭  
굵기 : 굵게

# 텍스트 스타일

- 글자 색 지정
  - color: <색상>
  - 색상 표현 방법은
    - 16진수 표기법(#RRGGBB)
    - rgb, rgba
    - hsl(hue, saturation, lightness), hsla
    - 색상 이름 표기법(aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow) 등 모두 216가지

<https://codepen.io/hajinsu/pen/YrwgPO>

Kakao

# 텍스트 줄 표시

- text-decoration: none | underline | overline | line-through

속성 값	설명
none	줄 표시 안함. 기본값
underline	밑줄 표시
overline	영역 위로 선을 표시
line-through	취소선 표시

<https://codepen.io/hajinsu/pen/yzewgg>

Kakao

# 텍스트 대/소문자 변환

- text-transform: none | capitalize | uppercase | lowercase | full-width

속성 값	설명
none	변환 없음
capitalize	시작하는 첫 글자를 대문자로 변환
uppercase	모든 글자를 대문자로 변환
lowercase	모든 글자를 소문자로 변환
full-width	가능한 모든 문자를 전각 문자로 변환

j a v a s c r i p t 를 공부해봅시다

javascript를 공부해봅시다

# 그림자 효과

- `text-shadow: none | <가로거리> <세로거리> <번짐 정도> <색상>`

속성 값	설명
<가로 거리>	텍스트로부터 그림자까지의 가로 길이. 양수는 오른쪽, 음수는 왼쪽
<세로 거리>	텍스트로부터 그림자까지의 세로 길이. 양수는 아래쪽, 음수는 위쪽
<번짐 정도>	그림자가 번지는 정도. 양수는 퍼져나감, 음수는 좁혀짐
색상	그림자 색상을 지정. 한가지만 지정할 수도 있고 공백으로 구분해 여러 색상을 지정할 수 있음

<https://goo.gl/ybkRRo>

# 공백 처리하기

- white-space: normal | nowrap | pre | pre-line | pre-wrap

속성 값	설명
normal	여러 개의 공백을 하나로 표시
nowrap	여러 개의 공백을 하나로 표시, 영역 너비를 넘어가는 내용을 한 줄로 표시
pre	여러 개의 공백을 그대로 표시, 영역 너비를 넘어가는 내용을 한 줄로 표시
pre-line	여러 개의 공백을 하나로 표시, 영역 너비를 넘어가는 내용은 자동으로 줄바꿈. 줄바꿈은 그대로 표시
pre-wrap	여러 개의 공백을 그대로 표시, 영역 너비를 넘어가는 내용은 자동으로 줄바꿈

<https://goo.gl/hGFVDy>



# 텍스트 간격 조절

- 글자 사이의 간격을 조절
  - letter-spacing: normal | <크기>
- 단어와 단어 사이의 간격을 조절
  - word-spacing: normal | <크기>

<https://goo.gl/8EY6L3>

# 문단 스타일

# 글자 쓰기 방향

- direction: ltr | rtl
- 한글이나 영어처럼 왼쪽에서 오른쪽으로 쓰는 언어일 경우, 속성값을 rtl로 입력해도 표시 순서는 바뀌지 않고 왼쪽에서 오른쪽으로 정렬되어 나타남

속성 값	설명
ltr	왼쪽에서 오른쪽으로 텍스트를 표시(left to right)
rtl	오른쪽에서 왼쪽으로 텍스트를 표시(right to left)

# 텍스트 정렬

- text-align: start | end | left | right | center | justify | match-parent

속성 값	설명
start	현재 텍스트 줄의 시작 위치에 맞추어 문단을 정렬. left-to-right 언어라면 왼쪽, right-to-left 언어라면 오른쪽에 맞춰 정렬
end	현재 텍스트 줄의 끝 위치에 맞추어 문단을 정렬. left-to-right 언어라면 오른쪽, right-to-left 언어라면 왼쪽에 맞춰 정렬
left	왼쪽에 맞춰 정렬
right	오른쪽에 맞춰 정렬
center	가운데에 맞춰 정렬
justify	양쪽에 맞춰 정렬
match-parent	부모 요소를 따라 정렬. 부모 속성의 값이 start, end일 경우 부모요소가 ltr인지 rtl인지에 따라 left나 right 값으로 계산해 적용

# 텍스트 들여쓰기

- text-indent: <크기> | <백분율>

속성 값	설명
<크기>	단위와 함께 들여 쓸 크기를 지정. 음수도 지정할 수 있음
<백분율>	부모 요소의 너비를 기준으로 상대적 크기를 지정

<https://goo.gl/m8rzVN>

# 줄 간격 조절

- line-height: normal | <숫자> | <크기> | <백분율> | inherit
- <https://goo.gl/NcsxQT>

# 넘치는 텍스트 표시

- text-overflow: clip | ellipsis
  - 조건
    - overflow: hidden | scroll | auto
    - white-space: nowrap
- <https://goo.gl/xg6eAY>

# 목록과 링크 스타일



# 목록의 불릿/번호 스타일

- `list-style-type: none` | <순서 없는 목록의 불릿> | <순서 목록의 번호>
- 순서 없는 목록
  - `disc`(채운 원), `circle`(빈 원), `square`(채운 사각형), `none`

# 목록의 불릿/번호 스타일

- 순서 목록

속성 값	설명	예시
decimal	1로 시작하는 십진수	1,2,3,...
decimal-leading-zero	앞에 0이 붙는 십진수	01,02,...,10,11
lower-roman	소문자 로마 숫자	i, ii, iii, iv, ...
upper-roman	대문자 로마 숫자	I, II, III, IV, ...
lower-alpha, lower-latin	소문자 알파벳	a, b, c, ...
upper-alpha, upper-latin	대문자 알파벳	A, B, C, ...

# 불릿 목록에서 이미지 사용

- list-style-image: <이미지> | none
  - <이미지> = url(파일 경로)

<https://goo.gl/uZYDkH>

# 목록에 들어 쓰는 효과 내기

- list-style-position: inside | **outside**
  - inside : 불릿이나 숫자를 안으로 들어 씀
  - outside : 불릿이나 숫자를 밖으로 내어 씀
- <https://goo.gl/mjMkuP>

# 실습 1

<https://goo.gl/xZdWZb>

## 최신 웹 디자인 트렌드

반응형 웹 디자인 - 다양한 화면 크기에 최적화하다

플랫 디자인 - 입체에서 평면으로

풀스크린 배경 - 콘텐츠에 집중

원 페이지 사이트 - 한 페이지에 모든 내용을 담다

패럴랙스 스크롤링 - 동적인 효과로 강한 인상을!

웹 폰트 - 웹 타이포그래피를 받쳐주는 기술

1. span 태그 사용
2. class 속성 사용
3. 텍스트 굵기 700
4. 굵은 텍스트의 크기는 부모 크기보다 한단계 크게 지정

# 실습 2

## 최신 웹 디자인 트렌드

**반응형 웹 디자인** - 다양한 화면 크기에 최적화하다

**플랫 디자인** - 입체에서 평면으로

**풀스크린 배경** - 콘텐츠에 집중

**원 페이지 사이트** - 한 페이지에 모든 내용을 담다

**패럴랙스 스크롤링** - 동적인 효과로 강한 인상을!

**웹 폰트** - 웹 타이포그래피를 받쳐주는 기술

1. 순서 없는 목록 사용
2. 불렛 없앰
3. 순서의 줄간격을 2배